

AMENDMENTS TO THE CLAIMS

- 1     1.     (currently amended) A method for dynamically allocating data buffers to a data  
2           structure, comprising the computer-implemented steps of:  
3           assigning a logging thread to said data structure, wherein said logging thread is  
4                           configured to insert free data buffers into said data structure;  
5           monitoring an amount of log data that is being stored within data buffers  
6                           associated with said data structure, wherein the step of monitoring the  
7                           amount of log data comprises the step of said logging thread tracking how  
8                           often said data buffers associated with said data structure are determined  
9                           to be full;  
10          based on the amount of log data that is being stored within said data buffers,  
11                           determining whether additional data buffers need to be linked into said  
12                           data structure; and  
13          if additional data buffers need to be linked to said data structure,  
14                           identifying one or more free buffers that are within a free buffer pool,  
15                           wherein said free buffer pool maintains free data buffers that may  
16                           be inserted into different data structures of a plurality of data  
17                           structures, each of which is associated with a different web site  
18                           domain; and  
19                           linking said one or more free data buffers into said data structure.
- 1     2.     (original) The method of Claim 1, further comprising the steps of:  
2           receiving requests for content that is associated with a web site domain;  
3           generating log data based on the requests; and  
4           writing said log data in one or more data buffers associated with said data  
5                           structure.

1 3. (original) The method of Claim 2, wherein the step of monitoring an amount of  
2 data that is being stored within data buffers includes the step of tracking how  
3 many requests are being received for content that is associated with said web site  
4 domain.

1 4. (original) The method of Claim 1, further comprising the steps of:  
2 determining that a particular data buffer should be removed from said data  
3 structure;  
4 unlinking said particular data buffer from said data structure; and  
5 inserting said particular data buffer into a ready-to-write buffer list.

1 5. (original) The method of Claim 4, wherein the step of determining that a  
2 particular data buffer should be removed comprises the step of detecting that said  
3 particular data buffer is full.

1 6. (original) The method of Claim 4, wherein the step of determining that a  
2 particular data buffer should be removed comprises the step of detecting that said  
3 particular data buffer has not been removed from said data structure for a  
4 particular period of time.

1 7. (currently amended) The method of Claim 4, further comprising the steps of:  
2 removing said particular data buffer from said ready-to-write buffer list, wherein  
3 said ready-to-write buffer list is located within a first memory area;  
4 storing log data information in said particular data buffer to a second memory  
5 area, wherein said second memory area is distinct from said first memory  
6 area; and  
7 inserting said particular data buffer into said[[a]] free buffer pool, ~~wherein said~~  
8 ~~free buffer pool maintains free data buffers that may be inserted into any~~

9                   ~~one of a plurality of data structures that are each associated with a~~  
10                   ~~particular web site domain.~~

1    8.   (original) The method of Claim 7, wherein:  
2           the step of identifying one or more free buffers comprises the step of selecting one  
3           or more free buffers from said free buffer pool; and  
4           the step of linking said one or more free data buffers into said data structure  
5           comprises the steps of,  
6           identifying one or more entries in said data structure; and  
7           linking said one or more free data buffers into said one or more entries in  
8           said data structure.

1    9.   (original) The method of Claim 4, wherein:  
2           said log data is generated based on request that are received for content associated  
3           with a particular web site domain; and  
4           said step of inserting said particular data buffer comprises the step of linking said  
5           particular data buffer into a queue that maintains only data buffers that  
6           contain log data associated with requests for said particular web site  
7           domain.

1    10-12. (canceled)

1    13.   (currently amended) A computer-readable medium carrying one or more  
2           sequences of instructions for dynamically allocating data buffers to a data  
3           structure, wherein execution of the one or more sequences of instructions by one  
4           or more processors causes the one or more processors to perform the steps of:  
5           assigning a logging thread to said data structure, wherein said logging thread is  
6           configured to insert free data buffers into said data structure;

7 monitoring an amount of log data that is being stored within data buffers  
8 associated with said data structure, wherein the step of monitoring the  
9 amount of log data comprises the step of said logging thread tracking how  
10 often said data buffers associated with said data structure are determined  
11 to be full;  
12 based on the amount of log data that is being stored within said data buffers,  
13 determining whether additional data buffers need to be linked into said  
14 data structure; and  
15 if additional data buffers need to be linked to said data structure,  
16 identifying one or more free buffers that are within a free buffer pool,  
17 wherein said free buffer pool maintains free data buffers that may  
18 be inserted into different data structures of a plurality of data  
19 structures, each of which is associated with a different web site  
20 domain; and  
21 linking said one or more free data buffers into said data structure.

1 14. (original) The computer-readable medium of Claim 13, further comprising  
2 instructions for performing the steps of:  
3 receiving requests for content that is associated with a web site domain;  
4 generating log data based on the requests; and  
5 writing said log data in one or more data buffers associated with said data  
6 structure.

1 15. (original) The computer-readable medium of Claim 14, wherein the step of  
2 monitoring an amount of data that is being stored within data buffers includes the  
3 step of tracking how many requests are being received for content that is  
4 associated with said web site domain.

- 1 16. (original) The computer-readable medium of Claim 13, further comprising  
2 instructions for performing the steps of:  
3 determining that a particular data buffer should be removed from said data  
4 structure;  
5 unlinking said particular data buffer from said data structure; and  
6 inserting said particular data buffer into a ready-to-write buffer list.
- 1 17. (original) The computer-readable medium of Claim 16, wherein the step of  
2 determining that a particular data buffer should be removed comprises the step of  
3 detecting that said particular data buffer is full.
- 1 18. (original) The computer-readable medium of Claim 16, wherein the step of  
2 determining that a particular data buffer should be removed comprises the step of  
3 detecting that said particular data buffer has not been removed from said data  
4 structure for a particular period of time.
- 1 19. (currently amended) The computer-readable medium of Claim 16, further  
2 comprising instructions for performing the steps of:  
3 removing said particular data buffer from said ready-to-write buffer list, wherein  
4 said ready-to-write buffer list is located within a first memory area;  
5 storing log data information in said particular data buffer to a second memory  
6 area, wherein said second memory area is distinct from said first memory  
7 area; and  
8 inserting said particular data buffer into said[[a]] free buffer pool, ~~wherein said~~  
9 ~~free buffer pool maintains free data buffers that may be inserted into any~~  
10 ~~one of a plurality of data structures that are each associated with a~~  
11 ~~particular web site domain.~~
- 1 20. (original) The computer-readable medium of Claim 19, wherein:

2 the step of identifying one or more free buffers comprises the step of selecting one  
3 or more free buffers from said free buffer pool; and  
4 the step of linking said one or more free data buffers into said data structure  
5 comprises the steps of,  
6 identifying one or more entries in said data structure; and  
7 linking said one or more free data buffers into said one or more entries in  
8 said data structure.

1 21. (original) The computer-readable medium of Claim 16, wherein:  
2 said log data is generated based on request that are received for content associated  
3 with a particular web site domain; and  
4 said step of inserting said particular data buffer comprises the step of linking said  
5 particular data buffer into a queue that maintains only data buffers that  
6 contain log data associated with requests for said particular web site  
7 domain.

1 22-24. (canceled)

1 25. (new) A method for processing requests for content that is associated with  
2 different web site domains, the method comprising:  
3 receiving, at a web server, a first request for access to first content that is  
4 associated with a first web site domain of a plurality of web site domains;  
5 queueing the first request within a connection queue;  
6 assigning a first server thread of a plurality of server threads to service the first  
7 request;

8 in response to being assigned a task of servicing the first request, the first server  
9 thread determining to which web site domain of the plurality of web site  
10 domains the first request is related;  
11 the first server thread loading first configuration data for the first web site domain  
12 in response to determining that the first request is related to the first web  
13 site domain, wherein, by loading the first configuration data, the first  
14 server thread is temporarily configured as a server thread that is dedicated  
15 to servicing requests for content that is available within the first web site  
16 domain;  
17 the first server thread generating first log data based on the first content, wherein  
18 the first log data includes information that identifies a first web page that  
19 was requested within the first web site domain;  
20 the first server thread selecting, from among a plurality of buffer files, a first  
21 buffer file that is associated with the first web site domain, wherein the  
22 first buffer file contains buffers that are to be used for buffering log data  
23 that is associated with the first web site domain;  
24 the first server thread selecting a first buffer from among a plurality of buffers  
25 within the first buffer file;  
26 the first server thread writing the first log data into the first buffer;  
27 a logging thread selecting, from among a plurality of physical log files, a first  
28 physical log file that is associated with the first buffer file, wherein each  
29 physical log file of the plurality of physical log files is associated with a  
30 separate buffer file of the plurality of buffer files;

31 a logging thread storing, into the first physical log file, information contained  
32 within the first buffer;  
33 receiving, at the web server, a second request for access to second content that is  
34 associated with a second web site domain of a plurality of web site  
35 domains, wherein the second web site domain is separate from the first  
36 web site domain;  
37 queueing the second request within the connection queue;  
38 assigning a second server thread of the plurality of server threads to service the  
39 second request, wherein the second server thread is separate from and  
40 executes concurrently with the first server thread;  
41 in response to being assigned a task of servicing the second request, the second  
42 server thread determining to which web site domain of the plurality of web  
43 site domains the second request is related;  
44 the second server thread loading second configuration data for the second web site  
45 domain in response to determining that the second request is related to the  
46 second web site domain, wherein, by loading the second configuration  
47 data, the second server thread is temporarily configured as a server thread  
48 that is dedicated to servicing requests for content that is available within  
49 the second web site domain, wherein the second configuration data differs  
50 from the first configuration data;  
51 the second server thread generating second log data based on the second content,  
52 wherein the second log data includes information that identifies a second  
53 web page that was requested within the second web site domain;



54 the second server thread selecting, from among the plurality of buffer files, a  
55 second buffer file that is associated with the second web site domain,  
56 wherein the second buffer file contains buffers that are to be used for  
57 buffering log data that is associated with the second web site domain,  
58 wherein the second buffer file differs from the first buffer file;  
59 the second server thread selecting a second buffer from among a plurality of  
60 buffers within the second buffer file;  
61 the second server thread writing the second log data into the second buffer;  
62 a logging thread selecting, from among the plurality of physical log files, a second  
63 physical log file that is associated with the second buffer file, wherein the  
64 second physical log file is separate from the first physical log file; and  
65 a logging thread storing, into the second physical log file, information contained  
66 within the second buffer.

1 26. (new) The method of Claim 25, further comprising:  
2 a logging thread determining whether to remove the first buffer from a buffer  
3 array that is contained in the first buffer file, wherein determining whether  
4 to remove the first buffer is based on at least one of (a) an amount of free  
5 space that is left within the first buffer and (b) an amount of time that the  
6 first buffer has been linked within the buffer array;  
7 in response to a logging thread determining that the first buffer should be  
8 removed from the buffer array, a logging thread moving the first buffer  
9 from the buffer array to a ready-to-write buffer list that is contained in the  
10 first buffer file.

1    27.    (new) The method of Claim 25, further comprising:  
2            a logging thread determining whether a current time of day is a specified time of  
3            day;  
4            in response to a logging thread determining that the current time of day is the  
5            specified time of day, a logging thread determining whether the first buffer  
6            contains log data; and  
7            in response to a logging thread determining that the first buffer contains log data  
8            as of the specified time of day, a logging thread moving the first buffer  
9            from a buffer array, which is contained in the first buffer file, to a ready-  
10          to-write buffer list that is contained in the first buffer file.